Are the BSDs dying? Some security researchers think so

Too few eyeballs on code is a security issue as vulnerabilities go unreported and unpatched. Can FreeBSD, OpenBSD, and NetBSD survive?

J.M. PorupBy J.M. Porup
Senior Writer, CSO | JAN 25, 2018 3:00 AM PST

17 open source table laptop group
Thinkstock

The open source Berkeley Software Distribution (BSD) versions of UNIX suffer from a lack of eyeballs on their code, and that hurts their security, Ilja van Sprundel, director of penetration testing at IOActive, told an audience at 34c3 in Leipzig, Germany at the end of December.

[Learn How to track and secure open source in your enterprise. | Get the latest from CSO by signing up for our newsletters.]

Struck by the small number of reported BSD kernel vulnerabilities compared to Linux, van Sprundel sat down last summer and reviewed BSD source code in his spare time. "How come there are only a handful of BSD security kernel bugs advisories released every year?" he wanted to know. Is it because the BSDs are so much more secure? Or is it because no one is looking?

van Sprundel says he easily found around 115 kernel bugs across the three BSDs, including 30 for FreeBSD, 25 for OpenBSD, and 60 for NetBSD. Many of these bugs he called "low-hanging fruit." He promptly reported all the bugs, but six months later, at the time of his talk, many remained unpatched.

"By and large, most security flaws in the Linux kernel don't have a long lifetime. They get found pretty fast," van Sprundel says. "On the BSD side, that isn't always true. I found a bunch of bugs that have been around a very long time." Many of them have been present in code for a decade or more.

OpenBSD the "clear winner" for security

OpenBSD's focus on security for the last two decades shows in the code, van Sprundel told the audience. "OpenBSD by far has the most knowledgeable developers when it comes to security."

For one thing, OpenBSD has a much smaller code base, around 2.9 million lines of code, compared to FreeBSD's roughly 9 million, and NetBSD's 7.3 million. "Obviously this plays a part," van Sprundel says. "You can't have a bug in code you don't have."

This smaller code base is partly by accident, he suggested, in that a lack of developer resources has prevented OpenBSD from implementing all the features they want, but also smaller on purpose, a deliberate decision to reduce attack surface, including removing support for rare, uncommon and older devices and architectures.

[Prepare to become a Certified Information Security Systems Professional with this comprehensive online course from PluralSight. Now offering a 10-day free trial!]

In terms of code quality, van Sprundel also praised OpenBSD code, noting that "low-hanging fruit like integer overflows are as good as gone in most places," and that "the most consistent quality was observed with OpenBSD."

However, OpenBSD's relative lack of popularity hurts the operating system's security, he suggested. "Bugs are still easy to find. If there were more people looking at OpenBSD, there would be more bugs [reported]."

Theo De Raadt, the founder of OpenBSD, agreed with van Sprundel that more eyeballs on OpenBSD would make the operating system more secure. "I remember reading his first slides, which were mostly about the impact of small API misuses," De Raadt tells CSO Online by email. "Unfortunately, this is a problem of the volume of code relative to manpower. Ensuring all code is 100 percent bug-free and handles all exceptional conditions is a rather difficult problem."

van Sprundel also praised OpenBSD's response to his bug findings, saying that De Raadt responded within a week, and OpenBSD patched the flaws within a few days.

"I communicated with Ilja from the start and got our team working on his discoveries," De Raadt writes. "We solved them all within a week or so and made patches available for the ones that were important. In my experience the only way to be proactive and responsive in a volunteer-driven software project is to never allow deferral of an issue to later. Problems must be handled ASAP to keep the interest in them up."

NetBSD the "clear loser" in terms of code quality

NetBSD's focus for many years has been to support the widest range of hardware possible. With this objective, however, comes the need to include a large quantity of legacy and binary compatibility (compat) code of varying degrees of quality, van Sprundel pointed out, and as a result "NetBSD seems to be less

consistent with security code quality."

The NetBSD response to van Sprundel's bug reports was both strikingly good and bad. On the one hand, he says, "They fixed virtually all bugs submitted, pretty much overnight!". On the other hand, those patches have yet to be shipped to users six months later. "Unless you run your own builds from recently checked-out code, your NetBSD machine is still vulnerable."

NetBSD developers corrected van Sprundel's account, noting that NetBSD 7.1.1, shipped on December 22, 2017, (a week before his talk at 34c3) contains patches to the security issues he found. "A lot of his findings were in the binary compatibility layers, and these aren't things that are going to cause a remote vulnerability anyway," Taylor R Campbell, a member of the NetBSD Foundation board of directors, says. "Someone would need access to the system anyway to run that code."

The large number of bugs van Sprundel found in NetBSD, and the project's sluggish response, raises red flag about the future of NetBSD. "NetBSD is practically dead," Patroklos Argyroudis, a security researcher at Census IT Security Works, whose work on BSD security van Sprundel cited in his talk, says. "In the past there were some companies that were trying to support it commercially, but I think they are long gone now."

Although NetBSD is a volunteer-driven open source project without any full-time developers, Campbell and David Maxwell, a former member of the NetBSD foundation board, are both confident Agryroudis's pessimism is unfounded. "Our primary goal is to have a core system with a clean architecture, then it becomes really easy to port to new platforms," Maxwell says. "We'll probably continue to be strong in the place we've been historically."

"We are also notoriously bad at marketing," Campbell adds.

FreeBSD, the "most technically advanced" of the BSDs
Long heralded for the performance of its network stack, FreeBSD is by far the most popular of the three big BSDs and finds a home at Netflix and WhatsApp, among others. "In modern perf tests, FreeBSD is on par with Linux or surpassing it a little," van Sprundel says. "Any place where you could deploy Linux, it's safe to say you could probably deploy FreeBSD. They are massively deployed in lots of places."

FreeBSD responded to the 30 kernel bugs in about a week and fixed a few in their source code repository. However, the software project released only a handful of advisories, and "the status of the rest is unknown at the moment," according to van Sprundel.

The FreeBSD project pushed back on van Sprundel's findings, however. "One of the issues we have is there's a large variety of issues that are being found but there are some issues that have no practical exploit," Ed Maste, director of project development at the FreeBSD Foundation, and member of the elected FreeBSD core team, says. "We've started treating some of these as just bugs and not as security issues."

The lack of developers hurts FreeBSD's security, not only in their ability to respond to bug reports, but also to implement new, industry-standard security features, Argyroudis suggests. "The most popular BSD, the most technically advanced, is FreeBSD, but they don't have as many developers [as Linux], and that basically means they are a bit behind in terms of security features."

Only recently has FreeBSD implemented preliminary support for ASLR in userland, Argyroudis says, and does not yet support KASLR. He also questions whether FreeBSD's network stack is still a killer feature.

"Maybe ten years back there was this notion that FreeBSD was better performance wise, that its network stack was much better, and other such things," Argyroudis says. "I'm not so sure if that's the case anymore. I would definitely be skeptical about that."

Maste disagrees. "We are able to do a phenomenal amount of work with a much smaller developer base, phenomenal both in terms of quantity and quality of work compared to Linux," he says. "The suggestion that our future is somehow hampered by a lack of developers is absolutely untrue."

Do FreeBSD kernel vulnerabilities affect OS X?
There's a lot of FreeBSD code in Mac OS X, and the FreeBSD security team coordinates disclosure with Apple, van Sprundel says. It remains unclear, however, how badly these reported vulnerabilities affect Apple laptops. The Darwin kernel has diverged sharply from the FreeBSD of 15 years ago, and OS X has received a great deal more scrutiny from security researchers over the years.

"When I submitted the bugs I had to the FreeBSD guys, they asked 'Do you mind if we send this to the guys at Apple?'" van Sprundel says. "So, the security team at Apple has this list of bugs. I have no idea how much of it applies to them. There's probably a couple of bugs that apply there."

Apple did not respond to our request for comment, and Maste declined to speculate, pointing out that only Apple would know the answer to that question. NetBSD's Maxwell is quick to point out that OS X includes

code from not just FreeBSD, but also NetBSD and OpenBSD.

Are the BSDs dying?
Popularity affects security, it turns out. More eyeballs on code means shorter bug lifetimes, and more developers means new security features reach users faster. The BSDs have lost the battle for mindshare to Linux, and that may well bode ill for the future sustainability of the BSDs as viable, secure operating systems.

"Say what you will about the people reviewing the Linux kernel code, there are simply orders of magnitude more of them," van Sprundel concludes. "Based on my result, code quality alone can't account for the discrepancy between the bug numbers (BSD vs. Linux)."

OpenBSD may be the most likely to survive, despite being far less popular than FreeBSD at the moment, Argyroudis suggests. "I see a greater chance for OpenBSD to survive because it has a more focused use case, and targets specific things. FreeBSD, I think it's much more difficult for it to survive than OpenBSD."

Measuring the popularity of the BSDs is difficult, however, Maste argues. "One of the challenges with trying to measure or quantify the popularity of FreeBSD or the other BSDs is that in a lot of cases it's used in applications or deployments that are not particularly visible," he says, such as appliances or products that build on derivatives of FreeBSD.

The permissive BSD license makes it even harder to quantify the popularity of the BSDs. "For end users, things like the license on the code may not matter much," NetBSD's Maxwell says, "but for the people who build embedded systems, for the people who are building products, the licensing of the code is very important."

Argyroudis remains pessimistic about the future of the BSDs. "I love the BSD code base," he says, "and I would love to be able to tell you different things, like how much more popular FreeBSD is and how easy it is going to be to survive against Linux. But unfortunately I don't think that's the case."

"I think it boils down to a lack of developers."